True or false? (unless otherwise specified)

1. Let $(\bullet)_2$ denote binary representation. It holds that
$$(0.1011)_2 + (0.0101)_2 = 1.$$
2. Let $(\bullet)_7$ denote base 7 representation. It holds that
$$(1000)_7 \times (2)_7 = (2000)_7.$$
3. Let $p \in \mathbf{N}$. The set $\{(b_0.b_1b_2 \dots b_{p-1})_2 : b_i \in \{0,1\}\} \subset \mathbf{R}$ contains 2^p distinct real numbers.
4. The machine epsilon of a floating point format is the smallest strictly positive number that can be represented exactly in the format.
5. Let ε denote the machine epsilon for the `Float64` format. Any $x \in \mathbf{R}$ such that $-\varepsilon < x < \varepsilon$ cannot be represented in the `Float64` format.
6. Machine multiplication is commutative, meaning that $a \mathbin{\widehat{*}} b = b \mathbin{\widehat{*}} a$ for any `Float64` point numbers a and b.
7. If `x` is a `Float16` and `y` is a `Float64` number, then the result of `x + y` is a `Float64` number.
8. The real number `0.0` can be represented exactly in the `Float32` format.
9. It holds that
$$(0.\overline{011})_2 = \frac{3}{4}.$$
10. In Julia, `Float64(x) == Float32(x)` is `true` if `x` is a rational number.
11. The value of the machine epsilon for `Float64` format is the same as for the `Float32` format.
12. The spacing (in absolute value) between successive double-precision (`Float64`) floating point numbers is always equal to the machine epsilon.
13. **All** the natural numbers can be represented exactly in the double precision floating point format `Float64`.
14. Machine addition in the `Float64` format is associative but not commutative.
15. In Julia, `Float64(.4) == Float32(.4)` evaluates to `true`.
16. **(Bonus)** In Julia `10 + eps() == 1` evaluates to `true`.

    Explain briefly why:

17. **(Bonus)** In Julia `sqrt(1 + eps()) == 1 + eps()` evaluates to `false`.

    Explain briefly why: